Project Report

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

Controllable Text Simplification

ANLP (Monsoon 2023)

Team:

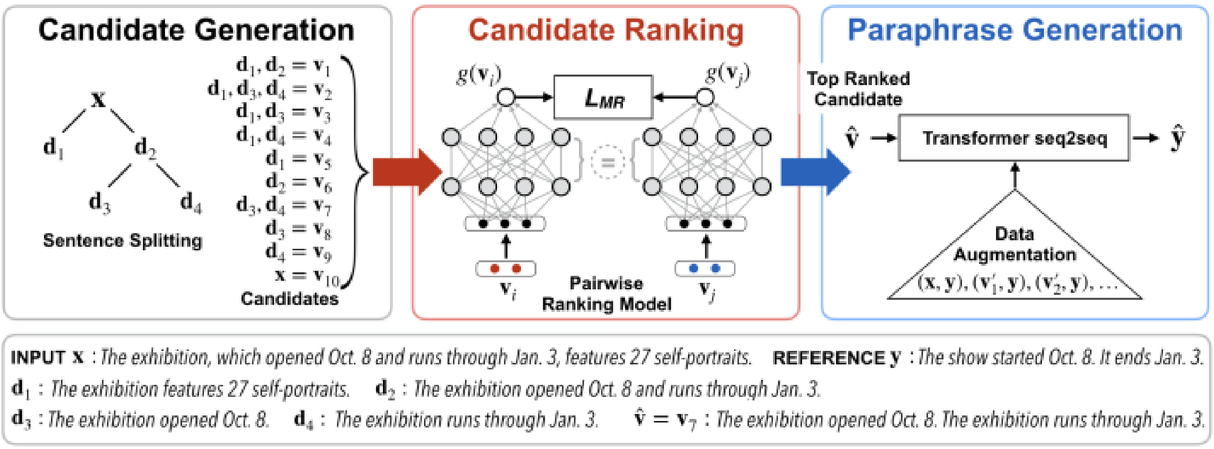
Aparna Agrawal	(2021121007)
Shreya Patil	(2021121009)
Adhiraj Deshmukh	(2021121012)

1. Introduction

Text simplification refers to the process of reducing the complexity of a piece of text while retaining its essential meaning. The primary goal of text simplification is to make text more accessible and understandable to a broader audience, including individuals with limited literacy skills, non-native speakers, people with cognitive impairments, or those who are reading in challenging situations. It also helps with downstream natural language processing tasks, such as parsing, semantic role labelling, information extraction, and machine translation.

2. Paper's Approach

Most recent text simplification systems have adopted sequence-to-sequence [12] models enhancing the fluency of their output. However, these models primarily rely on word deletion, often resulting in very short sentences that sacrifice the preservation of meaning [1]. While deleting words is a straightforward way to simplify sentences, it's not the most optimal or satisfying method. Human editors typically use a combination of techniques, including deletion, paraphrasing, and breaking sentences into smaller parts, to achieve better simplification.
As pointed out by Lee et al. in [4], another limitation of these end-to-end neural systems is the lack of control. To address these issues, Madela et al. [7] propose a novel hybrid approach that combines linguistically motivated syntactic rules with data-driven neural models to improve the diversity and controllability of the simplifications. We follow this approach in this project.



INPUT $\mathbf{x}$: *The exhibition, which opened Oct. 8 and runs through Jan. 3, features 27 self-portraits.* **REFERENCE** $\mathbf{y}$: *The show started Oct. 8. It ends Jan. 3.*
$\mathbf{d}_1$: *The exhibition features 27 self-portraits.* $\mathbf{d}_2$: *The exhibition opened Oct. 8 and runs through Jan. 3.*
$\mathbf{d}_3$: *The exhibition opened Oct. 8.* $\mathbf{d}_4$: *The exhibition runs through Jan. 3.* $\hat{\mathbf{v}} = \mathbf{v}_7$: *The exhibition opened Oct. 8. The exhibition runs through Jan. 3.*

Figure 1: over view of model

2.1 Step-1 : Sentence Splitting

- The paper uses rule based method + seq-to-seq model for splitting and deletion.
- The rule based method [9] uses 35 hand-Crafted English based rules for English based Standford parser called DisSim.
- We use a T5 model finetuned for text simplification as the seq-to-seq simplification model along with DisSim

2.2 Step-2 : Rank all the intermediate outputs(After Splitting and Deleting)

- A neural ranking model is used to score all the candidates that underwent splitting and deletion. The top-ranked candidate is then fed to the lexical paraphrasing model for the final output.
- To assess the "goodness" of each candidate v_i during training, a gold scoring function is defined as follows:

  $$g^*(v_i, y) = e^{-\lambda||\phi_{v_i} - \phi_y||} \times BERTScore(v_i, y)$$

  This is a length penalized BERTScore. λ is a hyperparameter which determines the extent of the length penalty.
- The ranking model is trained in a pairwise setup, comparing the candidates pairwise, and learning to minimize the ranking violations using hinge loss as follows:

  $$L_{MR} = \frac{1}{m} \sum_{k=1}^{m} \left[\frac{1}{n_k^2} \sum_{1=1}^{n_k} \left[\sum_{j=1, j \neq i}^{n_k} \max(0, 1 - l_{ij}^k d_{ij}^k) \right] \right]$$

  $$d_{ij}^k = g(v_i^k) - g(v_j^k)$$

  $$l_{ij}^k = sign\Big(g^*(v_i^k, y^k) - g^*(v_j^k, y^k)\Big)$$

  Here g refers to the neural model, which is a MLP, and L_{MR} is the loss to being minimized.

2.3 Step-3 : Paraphrase Generation Model

- This component is Transformer trained on data augmented data focused on Paraphrasing, to encourage diverse paraphrases.
- Data-Augmentation: - few selected candidates,in addition to the original input,are paired with human reference.
- Additional control over degree of paraphrasing: - A copy control token as Soft-Constraint. - a copy control token embedding is prepend to input embedding which control the proportion of the words need to be copied. - An auxiliary task(where a word should be copied) using a monolingual word aligner to derive noisy training labels.

3. Data

The paper primarily uses *Newsela* [10] dataset to train and evaluate the proposed model.

- **Newsela** [10]
  Newsela consists of $1,882$ news articles with each article rewritten by professional editors for students in different grades.
- **Newsela-AUTO** [3]
  Complex-simple sentence pairs from Newsela dataset, automatically aligned by Jiang et al. (2020)

As Newsela and Newsela-AUTO datasets are only available via request, we will be using the following corpora, for our training and evaluations:

- **Wiki-AUTO** [3]
  Complex-simple sentence pairs from Wikipedia corpus, which contains $138,095$ article pairs and $604k$ non-identical aligned and partially-aligned sentence pairs
- **TURK** [10] or **Wikipedia-TURK** corpus (Xu et al., 2016)
  - Used in testing and validation for lexical paraphrasing.
  - 8 human-written references for 2000 validation and 359 test sentences
- **ASSET** corpus [2]
  - Used in testing and validation for multiple rewrite operations.
  - 10 references for the same sentences

wiki_auto_asset_turk is a combination of the above three datasets made available on huggingface. We use a subset of this dataset to train and evaluate.

4. Metrics

- **SARI** [11]
  Averages the F1-precision of n-grams ($n \in \{1, 2, 3, 4\}$) inserted, deleted and kept when compared to human references.
- BLEU score with respect to the input (**s-BL**)
  - Just BLEU score is not reported because it often does not correlate with simplicity

5. Baselines

5.1 EditNTS

- EditNTS uses a model that learns to simplify sentences by predicting and executing explicit edit operations, such as adding, deleting, or keeping word
- The model is also more interpretable and controllable, as it provides edit traces and allows adjusting the edit operation ratios.
- It serves as a comparable baseline due to its focus on controllability.

5.2 DimSim

- DisSim applies 35 hand-crafted grammar rules to break down a complex sentence into a set of hierarchically organized sub-sentences.
- We've chosen this as one of our baselines as its state-of-the-art system for structural simplification, and due to its effectivness on complex sentences.

5.3 Pretrained Transformers

- Previous SOTA approaches include taking a pretrained checkpoint for Transformer Encoder and training the Decoder for text simplification tasks.
- Thus, we have considered taking Encoder-Decoder Transformer checkpoints such as BART and T5 as one of our baselines which are finetuned on text-simplification corpus such as Wiki-Auto.

5.4 Finetuning Transformers on DimSim Candidates

- We use the Pretrained Transformer checkpoints similar to the previous approach, but fine-tune them on the candidates generated by the DimSim module.
- We hypothesize that seq2seq generation model will learn lexical and structural paraphrases more efficiently as it offloads the burden of splitting and deletion to DimSim module, as it done in the main approach.

6. Implementational Details

6.1 Common Challenges

No reference code was available for this paper, thus we were unable to replicate some of the sections in the paper due to the ambiguity in their implementation details and lack of clear intuition behind those approaches. We suggest some alternatives and our motivation behind them in the upcoming parts for such sections.

6.2 Splitting and Deletion

DisSim: The code for DisSim was made available by the authors. The codebase uses the stanford-nlp module to split sentences using 35 hand-crafted rules.
Neural splitting: For neural splitting, we used the T5-small finetuned for text simplification that is made available as a part of huggingface.
For every input sentence, we constrain the beam search to generate 10 outputs with splitting and another 10 outputs without splitting. In interest of saving compute, 10 sentences are randomly chosen from these and are ranked.
Gaussian Binning: Gaussian binnning is a smooth binning approach and to project each numerical feature into a vector representation by applying multiple Gaussian radial functions. basis functions. For each feature f, we divide its value range $[f_{min}, f_{max}]$ evenly into k bins and place a Gaussian function for each bin with the mean $\mu_j(j \in 1, 2, ..., k)$ at the center of the bin and standard deviation σ. We specify σ as the width of the bin. The distance to each bin is computed as follows:

$$d_j(f(.)) = e^{\frac{(f(.)-\mu_j)^2}{2\sigma^2}}$$

teh vector corrosponding to the datapoint $f(i)$ is $f(.) = [d_j(f(1), d_j(\hat{f}(2))), \ldots, d_j(f(k))]$, where k is the number of bins.

Challenges and Limitations

- The code for DisSim was in Java, and had several dependency issues with the stanford-nlp module. Fixing this took longer than expected.
- Explaination of splitting in beam search was left ambiguous, and thus we interpreted them as beam search on split outputs from DimSim module.
- The paper proposes the use of Neural Splitting method to supplement more splitting and deletion based candidate simplified sentences, but there is no mechanism present while training this to ensure / incentivize the model to generate strictly splitting and deletion based generations. And thus it can freely paraphrase some parts of the sentence and we can't ensure meaning preservation using Jaccard Similarity, thus defeating the initial motivation to use this module.

6.3 Ranking

From the 20 candidates generated by the splitting and deletion module, 10 sentences are selected randomly to be ranked as candidates. To assess the "goodness" of each

candidate during training, we define the gold scoring function which is a length penalized BERTScore. The ranking model is trained in a pairwise setup. Margin Loss is used learn to minimize the pairwise ranking violations. The mathematical formulation of the gold scoring function and the loss function are described in 2.
The model used is a simple 4 layer feed-forward network, with a *tanh* activation function.
Features: For the feed-forward network, we use the following features:

- Number of words in the candidate.
- Compression ratio of the candidate with respect to the source.
- Jaccard similarity between the candidate and the source.

We vectorize all the real-valued features using Gaussian binning [8].

Challenges and Limitations

- The paper also uses the rules used for splitting and deletion in the sentence. Due to the code for DisSim being unreadable, we were unable to extract these features.
- No clear motivation behind hand-crafted features have been provided and leaves a scope of improvement in this part.

6.4 Paraphrasing

Copy Control: Given the input candidate, the percentage of copying $c \in [0, 1]$, the goal is to paraphrase the rest of $1 - c\%$ of the words to provide controllability over the simiplification of generated output.
To achieve this, paper converts c into a vector of the same dimension as word embeddings using Gaussian binning concatnate it to the beginning of the input sequence to the encoder. The Transformer encoder then produces a sequence of context-aware hidden states. These are then fed into the copy network which predicts the probability of each word being copied in the output. The hidden states are the n scled according to these probabilities, and passed to the decoder. The mathematical formulation of this is as follows:

$$h_1, h_2, \ldots, h_n = encoder(c; w_1, w_2, \ldots, w_n)$$

$$\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_n = (h_1 + p_1, h_2 + p_2, \ldots, h_n + p_n) \cdot u$$

Challenges and Limitations

- From the implementation details of the paper, it seems that while training, c_p value is set according to some distribution which is centered around a predefined arbitrary value between $(0, 1)$. Even if c_p values are changed at each epoch, the reference sentence would always remain the same.
- Which indicates that there is no-correlation between the value of c_p and the number of common words between candidates and reference while training.

- This deems c_p as a useless parameter and the model learn to represent anything using this information.
- This limitation steams from our interpretation of the limited and ambiguous details provided by the authors about this module. More details about the training task of the Binary Classifier would've cleared up some of the mis-interpretation, if it exists.

Proposed Alternate Approaches

- **Approach-1)** We take motivation from Martin et al. (LREC 2020) [6], which provides explicit control on simplification systems based on Sequence-to-Sequence models by discrete parametrization mechanism. And we represent the Copy control parameter similarly by dividing the value space $(0.0, 1.0]$ into multiple disceret buckets, which are represented by their corresponding Token vectors, concatenated to the input sequence.
- While training the token is selected using the ratio of number of common words between the input and reference sentences and total words, and then selecting the bucket corresponding to its value range.

- **Approach-2)** Another interesting approach that we encountered is Cripwell et al., (Findings 2022) [5] which provides explicit control over simplification via operation classification. They introduce a token which represents the class of the operation required for simplification of the input sentence.
- This is similar to the previous approach but we have less fine-grained control over the extent of the simplification caused by an operation of choice.
- Thus this doesn't serve as a direct alternative to the copy control module.
- This approach also requires curated operation labels of the paired sentences in the dataset for fine-tuning.

7. Analysis

7.1 Qualitative Results

DisSim:
SENTENCE:

- They would later return to the revived series in the 2008 Christmas Special "The Next Doctor", introducing two new variants of the race; the Cyber-Shades and the Cyber-King.

CANDIDATE SPLITS:

- They would later return to the revived series in the 2008 Christmas Special " The Next Doctor " .

- They was introducing two new variants of the race .
- They was introducing two new variants of the race .
- They was introducing the Cyber-Shades .
- They was introducing the Cyber-King .

Paraphrasing:

Models	**Output**
complex	Since,2010, project researchers have uncovered documents in portugal that have revealed who owned the ship.
Simple	Since 2010, experts have been figuring out who owned the ship.
Bart-finetuned	Since, 2010, project researchers have uncovered documents in portugal that have revealed who owned the ship.
Bart-controlled-finetuned	Experiments/ Comparison
T5-finetuned	Since 2010, project researchers have found documents in portugal that have revealed who owned the ship.
T5-controlled-finetuned	Since 2010, project researchers have uncovered documents in Portugal that have revealed who owned the ship.
Bart-controlled-finetuned-with-Roberta-classifer	Since,2010, project researchers have uncovered documents in Portugal that show who owned the ship.

Models	**Output**
complex	Experts say China's air pollution exacts a tremendous toll on human health.
Simple	China's air pollution is very unhealthy.
Bart-finetuned	Some people say China's air pollution has a great toll on human health.
Bart-controlled-finetuned	
T5-finetuned	Since 2010, project researchers have found documents in portugal that have revealed who owned the ship.
T5-Controlled-finetuned	Experts say China's air pollution has a huge impact on the health of the human body
Bart-controlled-finetuned-with-Roberta-classifer	Experts say China's air pollution has a huge impact on human health.

7.2 Quantitative Results

Models	**SARI**	**S-BLEU**
complex	51.58	100.00
Simple	99.13	37.12
disim	32.50	13.068
EDITNTS	35.4	69.00
Bart-finetuned	44.85	52.31
Disim-with-Bart-finetuned	48.82	66.22
Bart-controlled-finetuned	Experiments/ Comparison	21 Oct - 02 Nov
T5-finetuned	42.44	61.2
Disim-with-T5-finetuned	48.46	66.59
T5-controlled-finetuned(cp_0.5)	40.23	67.48
Bart-controlled-finetuned-with-Roberta-classifer(not trained by us)	46.25	71.36

$$\left[\left[\left[\left[\left[\right]\right]\right]\right]\right]$$

References

Bibliography

[1] Alva-Manchego, F., Bingel, J., Paetzold, G., Scarton, C., and Specia, L. Learning how to simplify from explicit labeling of complex-simplified text pairs. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (2017), Asian Federation of Natural Language Processing.

[2] Alva-Manchego, F., Martin, L., Bordes, A., Scarton, C., Sagot, B., and Specia, L. ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (2020), D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Association for Computational Linguistics.

[3] Jiang, C., Maddela, M., Lan, W., Zhong, Y., and Xu, W. Neural CRF model for sentence alignment in text simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (2020), D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Association for Computational Linguistics.

[4] Lee, J., and Yeung, C. Y. Personalizing lexical simplification. In *Proceedings of the 27th International Conference on Computational Linguistics* (2018), Association for Computational Linguistics.

[5] Liam Cripwell, J. L., and Gardent, C. Controllable sentence simplification via operation classification. In *Findings of the Association for Computational Linguistics: NAACL 2022* (2022), Association for Computational Linguistics.

[6] Louis Martin, Éric de la Clergerie, B. S., and Bordes., A. Controllable sentence simplification. In *Proceedings of the Twelfth Language Resources and Evaluation Conference* (2020), Association for Computational Linguistics.

[7] Maddela, M., Alva-Manchego, F., and Xu, W. Controllable text simplification with explicit paraphrasing. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (2021), Association for Computational Linguistics.

[8] Maddela, M., and Xu, W. A word-complexity lexicon and a neural readability ranking model for lexical simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (2018), E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds., Association for Computational Linguistics.

[9] Niklaus, C., Cetto, M., Freitas, A., and Handschuh, S. DisSim: A discourse-aware syntactic text simplification framework for English and German. In *Proceedings of the 12th International Conference on Natural Language Generation* (2019), K. van Deemter, C. Lin, and H. Takamura, Eds., Association for Computational Linguistics.

[10] Xu, W., Callison-Burch, C., and Napoles, C. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics 3* (2015).

[11] Xu, W., Napoles, C., Pavlick, E., Chen, Q., and Callison-Burch, C. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics 4* (2016).

[12] Zhang, X., and Lapata, M. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (2017), Association for Computational Linguistics.